

# A Management Framework for Automating Network Experiments and User Behaviour Emulation on Large Scale Testbed Facilities

Steven Latré<sup>1</sup>, Filip De Turck<sup>1</sup>, Dimitri Papadimitriou<sup>2</sup>, and Piet Demeester<sup>1</sup>

<sup>1</sup> Ghent University – IBBT, Department of Information Technology,  
Gaston Crommenlaan 8/201, B-9050, Gent, Belgium

<sup>2</sup> Alcatel-Lucent Bell Labs, Copernicuslaan 50, B-2018 Antwerpen, Belgium

**Abstract.** Generic test environments such as Emulab allow to perform large scale tests on different network topologies. While these facilities offer a tool to easily configure the topology, setting up realistic network scenarios afterwards is a manual and time consuming task involving the configuration of dozens of servers, including the installation of software suites and the emulation of subscriber behaviour. Also collecting the evaluation results afterwards can be complex and time consuming. This article discusses a management framework that allows both automating the configuration of networking experiments through a Graphical User Interface and automating the collection of measurements and visualisation of experimental results afterwards.

**Keywords:** experimental deployment, automation, validation

A large number of intelligent components for managing the Future Internet have been proposed recently or are currently being investigated. However, before these network components can be deployed in real-life networks, they need to be thoroughly validated through realistic and large scale experiments. Testbed facility management tools such as Emulab [1] provide a means to set up large scale network topologies but offer only a limited functionality in managing the deployment of the experiment itself. As such, they are only a first step towards easily deploying large scale tests. From past experience we have observed that in implementing and deploying an algorithm on a test environment, the implementation of the algorithm takes only a fraction of the time. Configuring and deploying this algorithm is a manual and error prone process that can easily take a few hours. In this paper, we propose a management framework which automates the configuration and management of network experiments.

The management framework allows setting up a complete test scenario, starting with configuring a topology, collecting measurements, and ending with collecting the evaluation results in only a few minutes. All this is configured in one Graphical User Interface (GUI) as illustrated in Figure 1, that communicates with the iLab.t Virtual Wall testbed facility. The framework supports 4 different functions, that correspond with the 4 phases of running a network experiment.

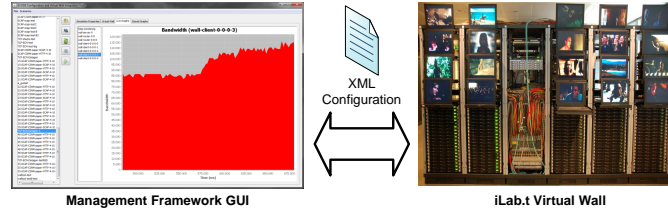


Fig. 1: Graphical User Interface for the management framework.

First, the tool allows configuring the experiment, both on the network and application level. This configuration consists of (i) defining an appropriate network topology, (ii) plugging in the corresponding network components and (iii) emulating realistic user behaviour that emulates how users access the available services. Additionally, the management framework features two simplification mechanisms that allow to reduce the complexity of the experiment both spatially (i.e. by grouping nodes onto one physical node) and temporally (i.e. by speeding up the experiment).

Second, the experiment can be started with one click of the button. This one-button deployment phase automatically interfaces with the iLab.t Virtual Wall facility and consists of otherwise time consuming tasks such as the configuration of the network topology, the reservation of the needed physical hardware in the Virtual Wall and the separate configuration of each physical node separately. This automated configuration reduces the experimental setup time to a couple of minutes instead of hours compared to manual (and thus potentially error-prone) configuration.

Third, the tool provides real-time visualization of the experiment as it progresses. By showing continuously updating graphs containing network statistics such as total bandwidth on each link, a clear view is provided on the status and the results of the experiment as it progresses.

The fourth and final stage consists of automatically collecting the training sets containing realistic traffic patterns that are constructed during the experiment. For example, for a machine learning specific scenario, clustering algorithms that are under investigation are applied to the newly constructed data set and the performance can be evaluated through the automated generation of graphs that provide visualization of the clustering. For more information about the architecture and performance of the management framework itself, we refer to [2].

## References

1. White, B., Lepreau, J., Stoller, L., Ricci, R., Guruprasad, S., Newbold, M., Hibler, M., Barb, C., Joglekar, A.: An integrated experimental environment for distributed systems and networks. In: In Proceedings of the Fifth Symposium on Operating Systems Design and Implementation. (2002) 255–270
2. Latré, S., Van de Meerssche, W., Melis, S., Papadimitriou, D., De Turck, F., Demeester, P.: Automated management of network experiments and user behaviour emulation on large scale testbed facilities. In: Proc. of the 6th International Conference on Network and Service Management (CNSM 2010). (2010)